

In the Claims:

Please add new claims 47-50. The claims are as follows:

1. (Previously presented) A method for document analysis and retrieval, comprising the following steps performed in the order recited:

transmitting, by a remote host in a first computing system to a web service host in a second computing system, a first portion of a document; and

sequentially transmitting, by the remote host to the web service host, at least one additional portion of the document, wherein the first portion and the at least one additional portion collectively comprise the entire document, wherein the entire document is adapted to be reconstructed and subsequently processed via processing said entire document by the web service host, said processing comprising at least one of:

extracting text from said entire document to configure said text in a text format, if said entire document received by said web service host comprises said text in a non-text format;

generating document keys associated with said text from analysis of said text in said text format, if said entire document received by said web service host comprises said text in said text format, or if said web service host has previously performed said extracting such that said text in said text format is available to said web service host; and

determining, from given categories of a document taxonomy, a set of closest categories to the document based on a comparison between the document keys and category keys of the given categories, if said entire document received by said web

service host comprises said document keys, or if said web service host has previously performed said generating such that said document keys are available to said web service host.

2. (Original) The method of claim 1, further comprising prior to the sending step identifying said web services host, said identifying comprising:

executing a Universal Description, Discovery, and Integration (UDDI) search to identify one or more web services hosts who can receive said document in chunks and who can perform said at least one of said extracting, generating, and stemming; and
selecting said web services host from said one or more web services hosts.

3. (Original) The method of claim 1, wherein said transmitting and sequentially transmitting comprise respectively transmitting and sequentially transmitting the first portion and the at least one additional portion via Internet transmission to said web service host.

4. (Original) The method of claim 1, wherein said generating comprises:

generating tokens of said text such that stop words do not appear in said tokens; and
stemming said tokens to generate said document keys from said tokens.

5. (Original) The method of claim 1, wherein said processing comprises said extracting, said generating, and said determining.

6. (Original) The method of claim 1, wherein said processing consists of two of said extracting, said generating, and said determining.

7. (Original) The method of claim 1, wherein said processing comprises said extracting but not said generating and not said determining.

8. (Original) The method of claim 1, wherein said processing comprises said generating but not said extracting and not said determining.

9. (Original) The method of claim 1, wherein said processing comprises said determining but not said extracting and not said generating.

10. (Previously presented) A system for document analysis and retrieval, comprising a first computing system that includes a remote host, wherein the remote host is remote relative to a web service host in a second computing system, and wherein the remote host is adapted to:

transmit a first portion of a document to the web service host; and

sequentially transmit at least one additional portion of the document to the web service host, wherein the first portion and the at least one additional portion collectively comprise the entire document, wherein the entire document is adapted to be reconstructed and subsequently processed via processing said entire document by the web service host, said processing comprising at least one of:

extracting text from said entire document to configure said text in a text format, if said entire document received by said web service host comprises said text in a non-text format; determine

generating document keys associated with said text from analysis of said text in said text format, if said entire document received by said web service host comprises said text in said text format, or if said web service host has previously performed said extracting such that said text in said text format is available to said web service host; and

determining, from given categories of a document taxonomy, a set of closest categories to the document based on a comparison between the document keys and category keys of the given categories, if said entire document received by said web service host comprises said document keys, or if said web service host has previously performed said generating such that said document keys are available to said web service host.

11. (Original) The system of claim 10, wherein the remote host is adapted to identify said web services host by:

executing a Universal Description, Discovery, and Integration (UDDI) search to identify one or more web services hosts who can receive said document in chunks and who can perform said at least one of said extracting, generating, and determining; and
selecting said web services host from said one or more web services hosts.

12. (Original) The system of claim 10, wherein to send transmit and to sequentially transmit comprises to respectively transmit and sequentially transmit the first portion and the at least one additional portion via Internet transmission to said web service host.

13. (Original) The system of claim 10, wherein said generating comprises:

generating tokens of said text such that stop words do not appear in said tokens; and
stemming said tokens to generate said document keys from said tokens.

14. (Original) The system of claim 10, wherein said processing comprises said extracting, said generating, and said determining.

15. (Original) The system of claim 10, wherein said processing consists of two of said extracting, said generating, and said determining.

16. (Original) The system of claim 10, wherein said processing comprises said extracting but not

said generating and not said determining.

17. (Original) The system of claim 10, wherein said processing comprises said generating but not said extracting and not said determining.

18. (Original) The system of claim 10, wherein said processing comprises said determining but not said extracting and not said generating.

19. (Previously presented) A method for document analysis and retrieval, comprising the following steps performed in the order recited:

receiving, by a web service host in a second computing system from a remote host in a first computing system, a first portion of a document;

sequentially receiving, by the web service host from the remote host, at least one additional portion of the document, wherein the first portion and the at least one additional portion collectively comprise the entire document;

reconstructing the entire document from the first portion and the at least one additional portion; and

processing the entire document by the web service host, wherein said processing comprises at least one of:

extracting text from said entire document to configure said text in a text format, if said entire document received by said web service host comprises said text in a non-text format;

generating document keys associated with said text from analysis of said text in said text format, if said entire document received by said web service host comprises said text in said text format, or if said web service host has previously performed said extracting such that said text in said text format is available to said web service host; and

determining, from given categories of a document taxonomy, a set of closest categories to the document, if said entire document received by said web service host comprises said document keys, or if said web service host has previously performed said generating such that said document keys are available to said web service host.

20. (Original) The method of claim 19, wherein the web services host is listed in a Universal Description, Discovery, and Integration (UDDI) registry as being able to receive said document in chunks and being able to perform said at least one of said extracting, generating, and determining.

21. (Original) The method of claim 19, wherein said receiving and sequentially receiving steps comprise receiving the first portion and the at least one additional portion via Internet transmission from said remote host.

22. (Original) The method of claim 19, wherein said generating comprises:
generating tokens of said text such that stop words do not appear in said tokens; and
stemming said tokens to generate said document keys from said tokens.

23. (Original) The method of claim 19, wherein said processing comprises said extracting, said generating, and said determining.

24. (Original) The method of claim 19, wherein said processing consists of two of said extracting, said generating, and said determining.

25. (Original) The method of claim 19, wherein said processing comprises said extracting but not said generating and not said determining.

26. (Original) The method of claim 19, wherein said processing comprises said generating but not said extracting and not said determining.

27. (Original) The method of claim 19, wherein said processing comprises said determining but not said extracting and not said generating.

28. (Original) The method of claim 19, wherein said determining comprises:

comparing the category keys of each category with said document keys to make a determination of a distance between the document and each category as a measure of how close the document is to each category; and

determining said set of closest categories based on said determination.

29. (Original) The method of claim 19, wherein said processing comprises said determining, and wherein the method further comprises:

creating a search string, said search string comprising a logical function of a subset of said document keys;

submitting said search string to a search engine;

receiving links to related documents from said search engine, said links being based on said search string; and

returning said links to said remote host.

30. (Previously presented) A system for document analysis and retrieval, comprising a second computing system that includes a web service host, wherein the web service host is remote relative to a remote host in a first computing system, and wherein the web service host is adapted to:

- receive a first portion of a document from the remote host;

- sequentially receive at least one additional portion of the document from the remote host, wherein the first portion and the at least one additional portion collectively comprise the entire document;

- reconstruct the entire document from the first portion and the at least one additional portion; and

- implement processing the entire document, said processing comprising at least one of:

 - extracting text from said entire document to configure said text in a text format, if said entire document received by said web service host comprises said text in a non-text format;

 - generating document keys associated with said text from analysis of said text in said text format, if said entire document received by said web service host comprises said text in said text format, or if said web service host has previously performed said extracting such that said text in said text format is available to said web service host; and

 - determining, from given categories of a document taxonomy, a set of closest categories to the document, if said entire document received by said web service host comprises said document keys, or if said web service host has previously performed said generating such that said document keys are available to said web service host.

31. (Original) The system of claim 30, wherein the web services host is listed in a Universal Description, Discovery, and Integration (UDDI) registry as being able to receive said document in chunks and being able to perform said at least one of said extracting, generating, and determining.

32. (Original) The system of claim 30, wherein to receive and sequentially receive comprise to receive the first portion and the at least one additional portion via Internet transmission from said remote host.

33. (Original) The system of claim 30, wherein said generating comprises:

generating tokens of said text such that stop words do not appear in said tokens; and
stemming said tokens to generate said document keys from said tokens.

34. (Original) The system of claim 30, wherein said processing comprises said extracting, said generating, and said determining.

35. (Original) The system of claim 30, wherein said processing consists of two of said extracting, said generating, and said determining.

36. (Original) The system of claim 30, wherein said processing comprises said extracting but not said generating and not said determining.

37. (Original) The system of claim 30, wherein said processing comprises said generating but not said extracting and not said determining.

38. (Original) The system of claim 30, wherein said processing comprises said determining but not said extracting and not said generating.

39. (Original) The system of claim 30, wherein said determining comprises:

comparing the category keys of each category with said document keys to make a determination of a distance between the document and each category as a measure of how close the document is to each category; and

determining said set of closest categories based on said determination.

40. (Original) The system of claim 30, wherein said processing comprises said determining, and wherein the method further comprises:

creating a search string, said search string comprising a logical function of a subset of said document keys;

submitting said search string to a search engine;

receiving links to related documents from said search engine, said links being based on said search string; and

returning said links to said remote host.

41. (Previously presented) The system of claim 1, wherein said determining comprises:

comparing the category keys of each category with said document keys to make a determination of a distance between the document and each category as a measure of how close the document is to each category; and

determining said set of closest categories based on said determination.

42. (Previously presented) The method of claim 41, wherein said comparing comprises computing said distance for each category as a dot product of a vector of the document keys and a vector of the category keys of each category.

43. (Previously presented) The system of claim 10, wherein said determining comprises:

comparing the category keys of each category with said document keys to make a determination of a distance between the document and each category as a measure of how close the document is to each category; and

determining said set of closest categories based on said determination.

44. (Previously presented) The system of claim 43, wherein said comparing comprises computing said distance for each category as a dot product of a vector of the document keys and a vector of the category keys of each category.

45. (Previously presented) The method of claim 28, wherein said comparing comprises computing said distance for each category as a dot product of a vector of the document keys and a vector of the category keys of each category.

46. (Previously presented) The system of claim 39, wherein said comparing comprises computing said distance for each category as a dot product of a vector of the document keys and a vector of the category keys of each category.

47. (New) The method of claim 1, wherein said sequentially transmitting at least one additional portion of the document is performed after said transmitting the first portion of the document has been performed.

48. (New) The method of claim 10, wherein the remote host is adapted to sequentially transmit the at least one additional portion of the document to the web service host after the remote host has transmitted the first portion of the document to the web service host.

49. (New) The method of claim 19, wherein said sequentially receiving at least one additional portion of the document is performed after said receiving the first portion of the document has been performed.

50. (New) The method of claim 30, wherein the web service host is adapted to sequentially receive the at least one additional portion of the document from the remote host after the web service host has received the first portion of the document from the remote host.

REMARKS

The Examiner rejected claims 1, 3-10, 12-19, 21-30 and 32-46 under 35 U.S.C. § 102(a) as allegedly being anticipated by Rosenschein *et al.* (USPN 6,519,631 B1).

The Examiner rejected claims 2, 11, 20 and 31 under 35 U.S.C. § 103(a) as allegedly being unpatentable over Rosenschein *et al.* in view of Mahmoud, "Registration and Discovery of Web Services Using JAXR with XML Registries such as UDDI and ebXML", June 2002.

Applicants respectfully traverse the claim objections, the § 102 and § 103 rejections with the following arguments.

35 U.S.C. § 102(a)

The Examiner rejected claims 1, 3-10, 12-19, 21-30 and 32-46 under 35 U.S.C. § 102(a) as allegedly being anticipated by Rosenschein *et al.* (USPN 6,519,631 B1).

Table 1 depicts the Examiner's interpretation of how various recited words and phrases claimed by Applicant are represented in Rosenschein.

Table 1

Claimed By Applicant	Represented in Rosenschein (as alleged by Examiner)
remote host in first computing system	servers 90, 92, 94
web server host in second computing system	server 30
document	data described in col. 8, lines 56-58 (i.e., data in small window adjacent to designated word on display 64 of apparatus 20)
first portion of document	data described in col. 8, lines 56-58
at least one additional portion of document	data described in col. 8, lines 56-58

To understand the Examiner's indicated interpretation of the "data" in Rosenschein, col. 8, lines 56-58 representing the "claimed "document", Applicants note that Rosenschein, col. 8, lines 56-58 recites: "Typically, a definition of the designated word, or other small quantity of **data** is shown in a small window, which opens adjacent to the **designated word** and closes automatically" (emphasis added).

For clarification, Applicants point to FIG. 2 of Rosenschein, wherein the small window (beginning with "FLOWER(N): ...") points to the designated word of "FLOWERS". See also, Rosenschein, col. 7, lines 31-35 which recites: "In FIG. 2, user 60 has designated the word

"flowers" with pointing device 66, by placing an arrow pointer on the word, and, for instance, right-clicking, to indicate to client 52 that additional information is desired about flowers." It should also be noted from Rosenschein, col. 7, lines 20-22 that "FIG. 2 is a sample output of display 64, generated during use of apparatus 20".

In summary, the Examiner has interpreted the claimed "document" as the "data" referred to in Rosenschein, col. 8, lines 56-58. Said "data" is the data appearing in the small window which opens adjacent to the designated word appearing on display 64 of apparatus 20.

Applicants respectfully contend that Rosenschein does not anticipate claims 1, 10, 19, and 30, because Rosenschein does not teach each and every feature of claims 1, 10, 19, and 30, as illustrated in the following examples in light of the Examiner's allegations in Table 1.

As a first example of why Rosenschein does not anticipate claims 1, 10, 19, and 30, Rosenschein does not teach the feature: "transmitting, by a remote host in a first computing system to a web service host in a second computing system, a first portion of a document; and sequentially transmitting, by the remote host to the web service host, at least one additional portion of the document, wherein the first portion and the at least one additional portion collectively comprise the entire document".

The Examiner argues: "*Rosenschein et al.* teach ... transmitting, by a remote host in a first computing system to a web service host in a second computing system (*see Fig. 1, Examiner interprets the steps of. (a) drawing "one or more context-indicating words ...from the body of text" and transmitting "with the designated word to server 30 ", (b) the server evaluating "the designated word in the context of the context-indicating words ", and (c) and transmitting "data*

from database 34 responsive to the evaluation" to be sequential steps.), by the remote host to the web service host, at least one additional portion of the document (see col. 8, lines 56-58, *Examiner interprets "the data" to be a document.*), wherein the first portion and the at least one additional portion collectively comprise the entire document”.

In response, Applicants respectfully contend that the Examiner has incorrectly interpreted the preceding feature of claims 1, 10, 19, and 30 in several ways.

A first way that the Examiner has incorrectly interpreted the preceding feature of claims 1, 10, 19, and 30 is that the Examiner incorrectly interprets the claimed “sequentially transmitting” as performing *any steps* sequentially. Applicants note that since “sequentially” modifies “transmitting”, claims 1, 10, 19, and 30 require that “transmitting” steps are to be performed sequentially. Applicants further note that since “sequentially transmitting” modifies “at least one additional portion of the document”, claims 1, 10, 19, and 30 require that if the “at least one additional portion of the document” comprises additional portions P1, P2, P3, ... then the additional portions P1, P2, P3, ... are transmitted in the sequence P1, P2, P3, ... The Examiner’s application of “sequentially transmitting” to steps (a) “drawing ...”, (b) “*the server evaluating ...*”, and (c) “*transmitting data from database 34 ...*” is incorrect, because the (a) “drawing ...” step and the (b) “*the server evaluating ...*” step do not perform “transmitting ... at least one additional portion of the document” as claimed.

A second way that the Examiner has incorrectly interpreted the preceding feature of claims 1, 10, 19, and 30 is that the Examiner incorrectly interprets the “first portion of a document” and the “at least one additional portion of the document” to each consist of the “data” representing claimed “document”, namely the “data” described in Rosenschein, col. 8, lines 56-

58 . In other words, the Examiner does not distinguish between the “first portion of a document” and the “at least one additional portion of the document”. Applicants respectfully contend that the language “additional portion” in the phrase “at least one **additional portion** of the document” requires that the “at least one additional portion of the document” be **in addition to** the “first portion of a document. Thus, the “first portion of a document” and the “at least one additional portion of the document” are required to be different portions of the document, which Rosenschein does not teach.

A third way that the Examiner has incorrectly interpreted the preceding feature of claims 1, 10, 19, and 30 is that the Examiner’s argument fails to recognize that the steps of “transmitting ... a first portion of a document and “transmitting ... at least one additional portion of the document” are claimed as distinct steps. However, the Examiner has interpreted the preceding steps as the same step, namely the same step of “transmitting data from database 34 responsive to the evaluation”. Thus, based on the Examiner’s argument, Rosenschein does not teach the steps of “transmitting ... a first portion of a document and “transmitting ... at least one additional portion of the document” as distinct steps in violation of the language of claims 1, 10, 19, and 30.

As a second example of why Rosenschein does not anticipate claims 1, 10, 19, and 30, Rosenschein does not teach the feature: “extracting text from said entire document to configure said text in a text format, if said entire document received by said web service host comprises said text in a non-text format”.

The Examiner argues: “*Rosenschein et al.* teach ... extracting text from said entire

document to configure said text in a text format, if said entire document received by said web service host comprises said text in a non-text format (*see col. 8, lines 45-52, Examiner interprets "an OCR algorithm determines the text" to mean: text is extracted by OCR from the entire document if the document is "a standard broadcast" image.*);”

In response, Applicants respectfully contend that the preceding language of claims 1, 10, 19, and 30 require said “text” to be comprised by the claimed document, and the Examiner alleges that the claimed “document” is the data appearing in the small window which opens adjacent to the designated word appearing on display 64 of apparatus 20. However, the text determined by the OCR algorithm is not extracted from the claimed document (i.e., from the data appearing in the small window which opens adjacent to the designated word appearing on display 64 of apparatus 20), but rather is extracted from the information containing the designated word on the display 64 of apparatus 20 (e.g., from the information containing “MOTHER’S DAY IS COMING SOON. DON’T FORGET TO SEND MOM FLOWERS”). See Rosenschein, col. 8, lines 45-48 (“The user points to and clicks on the pitcher's name, and an OCR algorithm determines the text, which is transmitted to server 30 for retrieval therefrom of information related to the pitcher's name”). In other words, Rosenschein does not teach that the OCR algorithm extracts text from the “document” (i.e., from the data appearing in the small window which opens adjacent to the designated word appearing on display 64 of apparatus 20), as required by claims 1, 10, 19, and 30.

Moreover, it is clear from that the preceding quote in Rosenschein, col. 8, lines 45-48 that the text (i.e., “the pitcher’s name”) determined by the OCR algorithm is the “designated word”, particularly in light of Rosenschein, col. 7, lines 30-34 which recites: “In FIG. 2, user 60 has

designated the word "flowers" with pointing device 66, by placing an arrow pointer on the word, and, for instance, right-clicking, to indicate to client 52 that additional information is desired about flowers.”

Therefore, the Examiner’s interpretation of “text” is inconsistent with the language of claims 1, 10, 19, and 30

In addition, Applicants maintain that Rosenschein does not teach that said “extracting” is performed conditionally “if said entire document received by said web service host comprises said text in a non-text format”. The Examiner’s statement that “Examiner interprets "an OCR algorithm determines the text" to mean: text is extracted by OCR from the entire document if the document is "a standard broadcast" image” is an arbitrary interpretation not supported by anything disclosed in Rosenschein. Applicants maintain that the Examiner cannot successfully assert such an interpretation without supplying evidentiary support accompanied by competent analysis.

Accordingly, Rosenschein does not teach the preceding feature of claims 1, 10, 19, and 30.

As a third example of why Rosenschein does not anticipate claims 1, 10, 19, and 30, Rosenschein does not teach the feature: “generating document keys associated with said text from analysis of said text in said text format, if said entire document received by said web service host comprises said text in said text format, or if said web service host has previously performed said extracting such that said text in said text format is available to said web service host”.

The Examiner argues: “*Rosenschein et al.* teach ... generating document keys associated

with said text from analysis of said text in said text format, if said entire document received by said web service host comprises said text in said text format, or if said web service host has previously performed said extracting such that said text in said text format is available to said web service host (*see col. 8, lines 61-66, Examiner interprets "the designated word in the context of the context-indicating words " to be a key associated with said text"*).

In response, Applicants respectfully contend, as explained *supra* in conjunction with said second example, that the Examiner's interpretation of "text" is inconsistent with the language of claims 1, 10, 19, and 30.

In addition, the Examiner alleges that the "designated word" is a key associated with said "text". However, the Examiner also alleges that said "text" is what is determined by the OCR algorithm, which must be the "designated word" as explained *supra*. Therefore, the Examiner is effectively alleging that the designated word is a key associated with the designated word, which makes no sense at all.

Accordingly, Rosenschein does not teach the preceding feature of claims 1, 10, 19, and 30.

As a fourth example of why Rosenschein does not anticipate claims 1, 10, 19, and 30, Rosenschein does not teach the feature: "determining, from given categories of a document taxonomy, a set of closest categories to the document based on a comparison between the document keys and category keys of the given categories, if said entire document received by said web service host comprises said document keys, or if said web service host has previously performed said generating such that said document keys are available to said web service host".

The Examiner argues: “*Rosenschein et al.* teach ... determining, from given categories of a document taxonomy, a set of closest categories to the document based on a comparison between the document keys and category keys of the given categories, if said entire document received by said web service host comprises said document keys, or if said web service host has previously performed said generating such that said document keys are available to said web service host (*see col. 9, line 16 to col. 10, line 50, Examiner interprets "concept c_j " to be a category.*)”

In response, Applicants respectfully contend that the claimed “categories” are “categories of a document taxonomy”. However, *Rosenschein* does not teach that the concepts c_j are “categories of a document taxonomy” as required by claims 1, 10, 19, and 30. “Taxonomy” is defined as “classification, esp. in relation to its principles or laws” or “that department of science, or of a particular science, which deals with classification”. See “The American College Dictionary” 1242 (1955). Thus, in order for the concepts c_j to be categories of a taxonomy, the concepts c_j must relate to each other in accordance with a classification, which *Rosenschein* does not teach.

In addition, Applicants maintain that *Rosenschein* does not teach the specifics of said “determining”. In particular, *Rosenschein* does not teach determining a set of closest categories to the document “based on a comparison between the document keys and category keys of the given categories”. The Examiner considers s_1, s_2, \dots, s_n (discussed in *Rosenschein*, col. 9, line 58 - col. 10, line 50) to be the document keys. The Examiner’s allegation that the concepts c_j ($j=1, 2, \dots, N$) are the categories implies that the keywords k_1, k_2, \dots, k_n are the category keys. However, *Rosenschein* does not teach that the document keys s_1, s_2, \dots, s_n are compared with

category keys k_1, k_2, \dots, k_n to determine a set of closest categories. Rather, the category weight matrix elements W_{ij} are utilized, in combination with the modified positional weights p_1, p_2, \dots, p_N , to compute the score $S(c_j)$ in the equation depicted in Rosenschein, col. 10, lines 42-44.

Accordingly, Rosenschein does not teach the preceding feature of claims 1, 10, 19, and 30.

Based on the preceding arguments, Applicants respectfully maintain that Rosenschein does not anticipate claims 1, 10, 19, and 30, and that claims 1, 10, 19, and 30 are in condition for allowance. Since claims 3-9, 41 and 42 depend from claim 1, Applicant contends that claims 3-9, 41 and 42 are likewise in condition for allowance. Since claims 12-18, 43 and 44 depend from claim 10, Applicant contends that claims 12-18, 43 and 44 are likewise in condition for allowance. Since claims 21-29 and 45 depend from claim 19, Applicant contends that claims 21-29 and 45 are likewise in condition for allowance. Since claims 32-40 and 46 depend from claim 30, Applicant contends that claims 32-40 and 46 are likewise in condition for allowance.

In addition with respect to claims 5, 14, 23, and 34, Applicants respectfully contend that Rosenschein does not teach the feature: “wherein said processing comprises said extracting, said generating, and said determining”, based on the arguments presented *supra* in conjunction with claims 1, 10, 19, and 30, wherein said arguments presented *supra* explain that Rosenschein: does not teach said extracting, does not teach said generating, and does not teach said determining.

In addition with respect to claims 6, 15, 24, and 35, Applicants respectfully contend that

Rosenschein does not teach the feature: “wherein said processing consists of two of said extracting, said generating, and said determining”.

The Examiner argues: “*Rosenschein et al.* teach the method and system, wherein said processing can comprise some combination of the seven (7) out of eight (8) possible processing combinations, where processing comprises at least one of extracting, generating, and determining (see col. 12, lines 34-39).”.

In response, Applicants respectfully contend that the preceding argument by the Examiner is not persuasive, because the Examiner has not cited specifically where Rosenschein teaches “wherein said processing consists of two of said extracting, said generating, and said determining”.

In addition with respect to claims 7, 16, 25, and 36, Applicants respectfully contend that Rosenschein does not teach the feature: “wherein said processing comprises said extracting but not said generating and not said determining”.

The Examiner argues: “*Rosenschein et al.* teach the method and system, wherein said processing can comprise some combination of the seven (7) out of eight (8) possible processing combinations, where processing comprises at least one of extracting, generating, and determining (see col. 12, lines 34-39).”.

In response, Applicants respectfully contend that the preceding argument by the Examiner is not persuasive, because the Examiner has not cited specifically where Rosenschein teaches “wherein said processing comprises said extracting but not said generating and not said determining”.

In addition with respect to claims 8, 17, 26, and 37, Applicants respectfully contend that Rosenschein does not teach the feature: “wherein said processing comprises said generating but not said extracting and not said determining”.

The Examiner argues: “*Rosenschein et al.* teach the method and system, wherein said processing can comprise some combination of the seven (7) out of eight (8) possible processing combinations, where processing comprises at least one of extracting, generating, and determining (see col. 12, lines 34-39).”.

In response, Applicants respectfully contend that the preceding argument by the Examiner is not persuasive, because the Examiner has not cited specifically where Rosenschein teaches “wherein said processing comprises said generating but not said extracting and not said determining”.

In addition with respect to claims 9, 18, 27, and 38, Applicants respectfully contend that Rosenschein does not teach the feature: “wherein said processing comprises said determining but not said extracting and not said generating”.

The Examiner argues: “*Rosenschein et al.* teach the method and system, wherein said processing can comprise some combination of the seven (7) out of eight (8) possible processing combinations, where processing comprises at least one of extracting, generating, and determining (see col. 12, lines 34-39).”.

In response, Applicants respectfully contend that the preceding argument by the Examiner is not persuasive, because the Examiner has not cited specifically where Rosenschein teaches

“wherein said processing comprises said determining but not said extracting and not said generating”.

In addition with respect to claims 28 and 39, Applicants respectfully contend that Rosenschein does not teach the feature: “comparing the category keys of each category with said document keys to make a determination of a distance between the document and each category as a measure of how close the document is to each category; and determining said set of closest categories based on said determination”.

The Examiner argues: “Regarding claims 28 & 39. *Rosenschein et al.* teach the method and system, wherein said determining comprises: comparing the category keys of each category (*see col. 9, lines 16-57, Examiner interprets "concepts c_1, c_2, \dots, c_M " to be categories and keywords k_1, k_2, \dots, k_N to be the keys of each category.*) with said document keys (*see col. 9, lines 58-61, Examiner interprets s_1, s_2, \dots, s_N to be document keys.*) to make a determination of a distance between the document and each category as a measure of how close the document is to each category (*see col. 10, lines 1-45, Examiner interprets the score $S(c_j)$ to measure how close a concept (i.e., category) is to the current document.*); and determining said set of closest categories based on said determination (*see col. 10, lines 47-49, Examiner interprets the set of sorted scores, $S(c_j)$, to be the set of categories ordered by closeness.*)”.

In response, Applicants respectfully contend that the preceding argument by the Examiner is not persuasive, because Rosenschein does not teach that the document keys s_1, s_2, \dots, s_n are compared with category keys k_1, k_2, \dots, k_n to make a determination of a distance between the document and each category as a measure of how close the document is to each category.

Rather, the category weight matrix elements W_{ij} are utilized, in combination with the modified positional weights p_1, p_2, \dots, p_N , to compute the score $S(c_j)$.

In addition with respect to claims 29 and 40, Applicants respectfully contend that Rosenschein does not teach the feature: “creating a search string, said search string comprising a logical function of a subset of said document keys; submitting said search string to a search engine; receiving links to related documents from said search engine, said links being based on said search string”.

The Examiner argues: “Regarding claims 29 & 40. *Rosenschein et al.* teach the method and system, wherein said processing comprises said determining, and wherein the method further comprises: creating a search string, said search string comprising a logical function of a subset of said document keys (*see col. 9, lines 59-61, Examiner interprets $s = s_1, s_2, \dots, s_f \dots s_n$ to be a search string where the logical function (“s and f”) maps $s_1, s_2, \dots, s_f \dots s_n$ to $((s_1, s_2, \dots, s_f \dots s_n), f)$*); submitting said search string to a search engine (*see col. 9, lines 59-61*); receiving links to related documents from said search engine, said links being based on said search string (*see col. 5, lines 24-25, Examiner interprets “computer data relating to the at least one transmitted word” to comprise links.*)”

In response, Applicants respectfully contend that the preceding argument by the Examiner is not persuasive based on the following argument. The Examiner argues that Rosenschein, col. 9, lines 59-61 teaches a search engine to which the alleged search keys $s_1, s_2, \dots, s_f \dots s_n$ are submitted. By so citing Rosenschein, col. 9, lines 59-61, the Examiner is alleging that the context-determination algorithm (disclosed in Rosenschein, col. 9, lines 59-61) is a search

engine.

In response, Applicants maintain that that the Examiner has not supplied evidence with supporting argumentation to demonstrate that the context-determination algorithm is a search engine. However, even if the context-determination algorithm is a search engine, then Rosenschein does not teach “receiving links to related documents from said search engine, because the Examiner’s citation to Rosenschein, col. 5, lines 24-25 does not teach that the alleged links (namely, “computer data relating to the at least one transmitted word”) are received from the search engine (i.e., from the context-determination algorithm). Furthermore, the Examiner’s citation to Rosenschein, col. 5, lines 24-25 does not teach “said link being based on said search string” (i.e., the computer data relating to the at least one transmitted word being based on $s_1, s_2, \dots, s_f \dots s_n$).

In addition with respect to claims 28, 39, 41, and 43, Applicants respectfully contend that Rosenschein does not teach the feature: “comparing the category keys of each category with said document keys to make a determination of a distance between the document and each category as a measure of how close the document is to each category”.

The Examiner’s argument depends on the following allegation by the Examiner:
“Examiner provides Official Notice that the dot product of two vectors is a determination of the distance between the two vectors.”

In response, Applicants respectfully contend that the preceding allegation of Official Notice by the Examiner is incorrect as may be seen in the following examples.

In two-dimensional Euclidean space, the distance between two vectors which are parallel

to each other is independent of their dot product. Generally, in two-dimensional Euclidean space, the dot product $\underline{A} \cdot \underline{B}$ of vectors \underline{A} and \underline{B} is equal to $|\underline{A}| |\underline{B}| \cos(\underline{A}, \underline{B})$. Thus in two-dimensional Euclidean space, $\underline{A} \cdot \underline{B}$ may be interpreted as a measure of the projection of \underline{A} onto \underline{B} , or of the projection of \underline{B} onto \underline{A} , and has no relationship to a “distance” between \underline{A} and \underline{B} .

In three-dimensional Euclidean space, the distance between two vectors which do not intersect is a constant, and the dot product between the two vectors varies with the angle between the two vectors which do not intersect.

For two vectors which intersect each other in two-dimensional or in three-dimensional Euclidean space, the concept of a “distance” between the two vectors is meaningless and thus undefined.

Since, the Examiner’s argument depends on an erroneous allegation of Official Notice, Applicants respectfully maintain that Rosenschein does not teach the preceding feature of claims 28, 39, 41, and 43.

Furthermore, Applicants are not claiming that a dot product of two vectors is a determination of the distance between the two vectors. Therefore, the Examiner’s allegation of Official Notice, even if not erroneous, would not be persuasive as to the rejection of claims 28, 39, 41, and 43.

In addition with respect to claims 42, 44, 45, and 46, Applicants respectfully contend that Rosenschein does not teach the features: “comparing the category keys of each category with said document keys to make a determination of a distance between the document and each category as a measure of how close the document is to each category; and determining said set of closest

categories based on said determination ..., wherein said comparing comprises computing said distance for each category as a dot product of a vector of the document keys and a vector of the category keys of each category.”

The Examiner’s argument depends on the following allegation by the Examiner:

“Examiner interprets the modified positional weights, p_i to be a vector of document keys and each column, of W_{ij} to be a vector of the category keys of each category”.

In response, Applicants respectfully contend that the preceding argument by the Examiner is not persuasive, because;

(1) Claims 42, 44, 45, and 46 respectively depend from claims 1, 10, 28, and 39 and therefore comprise all limitations in claims 1, 10, 28, and 39. Since the Examiner’s analysis of claims 1, 10, 28, and 39 has asserted that the document keys are s_1, s_2, \dots, s_n and the category keys are k_1, k_2, \dots, k_n , the Examiner’s assertion that p_i is a vector of document keys and that each column of W_{ij} is a vector of the category keys has introduced a logical inconsistency with respect to antecedent basis (referring back to claims 1, 10, 28, and 39) and therefore demonstrates that the Examiner’s argument is logically flawed.

(2) Rosenschein does not teach that p_i is a vector of document keys and that each column of W_{ij} is a vector of the category keys. The Examiner’s assertion that p_i is a vector of document keys and that each column of W_{ij} is a vector of the category keys is arbitrary and is not supported by anything disclosed in Rosenschein. Applicants maintain that the Examiner cannot successfully assert such an interpretation without supplying evidentiary support accompanied by competent analysis.

35 U.S.C. § 103(a)

The Examiner rejected claims 2, 11, 20 and 31 under 35 U.S.C. § 103(a) as allegedly being unpatentable over Rosenschein *et al.* in view of Mahmoud, “Registration and Discovery of Web Services Using JAXR with XML Registries such as UDDI and ebXML”, June 2002.

Since claims 2, 11, 20 and 31 respectively depend from claims 1, 10, 19, and 30, which Applicants have argued *supra* to not be unpatentable over Rosenschein under 35 U.S.C. §102(b), Applicants maintain that claims 2, 11, 20 and 31 are likewise not unpatentable over Rosenschein in view of Mahmoud under 35 U.S.C. §103(a).

In addition, Rosenschein in view of Mahmoud does not teach or suggest the feature: “executing a Universal Description, Discovery, and Integration (UDDI) search **to identify one or more web services hosts who can receive said document in chunks and who can perform said at least one of said extracting, generating, and stemming**”.

The Examiner argues: “Examiner asserts that receiving documents in chunks, extracting, generating, and stemming text are services that can be published for a organization **by modifying** Code Sample 1: PublishORG.java” (emphasis added).

Thus, the Examiner admits that Mahmoud does not teach or suggest the preceding feature claims 2, 11, 20 and 31, since the Examiner asserts that the preceding feature claims 2, 11, 20 and 31 can be realized only through a **modification** of Code Sample 1: PublishORG.java in Mahmoud. Therefore, the Examiner’s argument is not persuasive due to lack of disclosure and/or lack of enablement. In particular, neither Rosenschein nor Mahmoud teaches or suggests the preceding feature claims 2, 11, 20 and 31, and/or the Examiner has not cited any prior art to

demonstrate how the alleged modification of Code Sample 1: PublishORG.java in Mahmoud would be enabled.